M.A.D.N.E.S.S.

Data Models and Formats, Visualization and Analysis

Jeremy Meredith

Spring 2010 VACET AHM

E. WES BETHEL (LBNL), CHRIS JOHNSON (UTAH), KEN JOY (UC DAVIS), SEAN AHERN (ORNL), VALERIO PASCUCCI (LLNL), JONATHAN COHEN (LLNL), MARK DUCHAINEAU (LLNL), BERND HAMANN (UC DAVIS), CHARLES HANSEN (UTAH), DAN LANEY (LLNL), PETER LINDSTROM (LLNL), JEREMY MEREDITH (ORNL), GEORGE OSTROUCHOV (ORNL), STEVEN PARKER (UTAH), CLAUDIO SILVA (UTAH), XAVIER TRICOCHE (UTAH), ALLEN SANDERSON (UTAH), HANK CHILDS (LLNL)

www.vacet.org

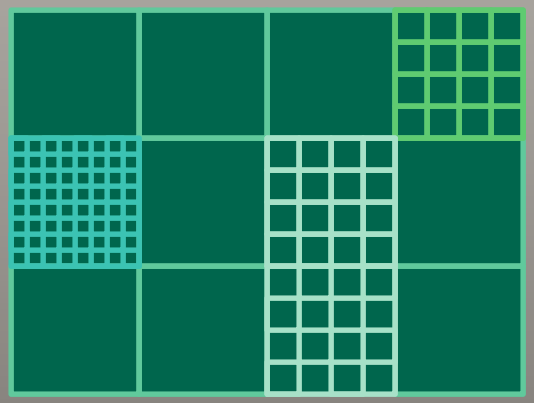
M.A.D.N.E.S.S.

- "Multiresolution ADaptive Numerical Environment for Scientific Simulation"
  - Support from multiple DOE, Office of Science projects (e.g. SciDAC)
- Scientific code framework
  - Primarily computational chemistry, nuclear
  - Extensible to other domains (computational fluid dynamics)

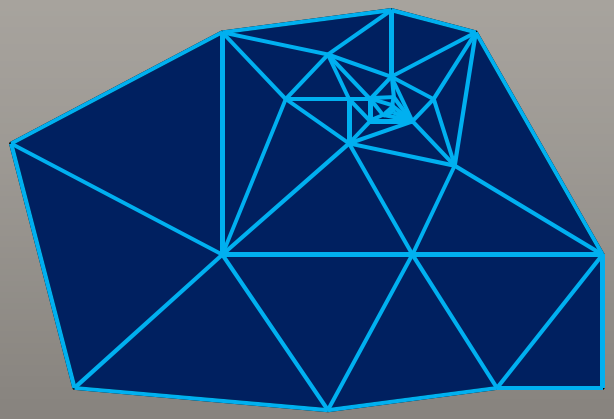
M.A.D.N.E.S.S.

- P.I. – Robert Harrison
  - largest OLCF INCITE allocation on Jaguar
- Main Problem
  - have (almost) no visualization capability
    - sampling to regular grids can make some pictures
  - no debugging, introspection, analysis capability
- Primary Obstacles
  - data model (meshes, dimensions)
  - data storage (not writing internal data)

VisIt supports varying resolution meshes

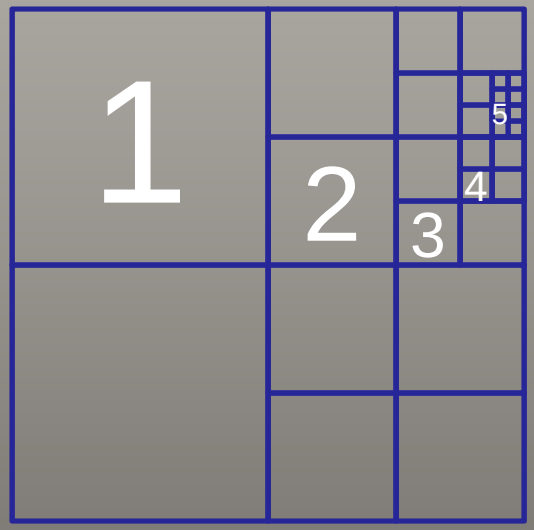


block-structured adaptive mesh refinement (AMR)

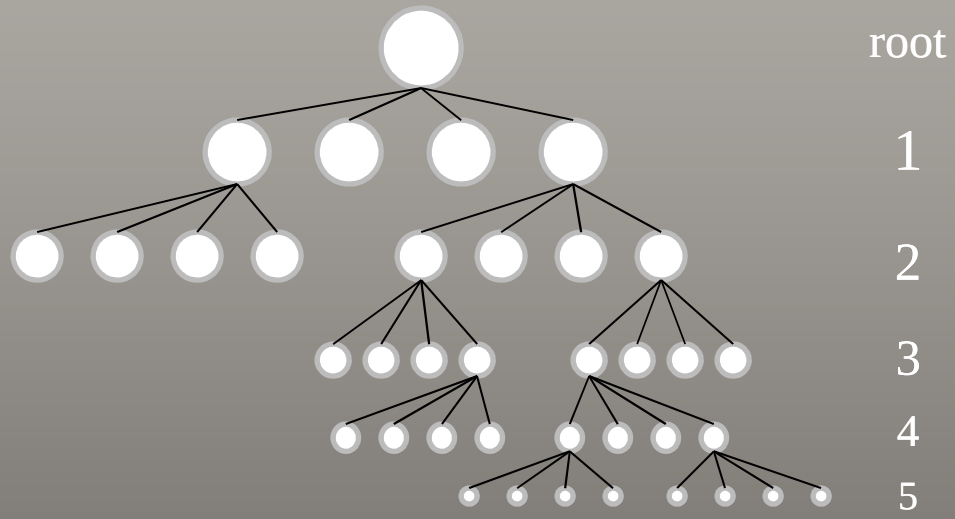


unstructured grid with non-uniform spatial resolution

MADNESS has no mesh, just a quadtree with polynomial coefficients



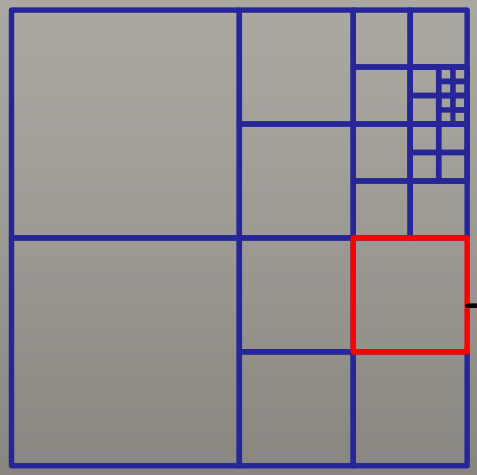
spatial structure



internal tree representation

- Up to 30 refinement levels / tree depth
- to 400 (2D) / 8000 (3D) coefficients per tree node

Legendre polynomial series at each tree node

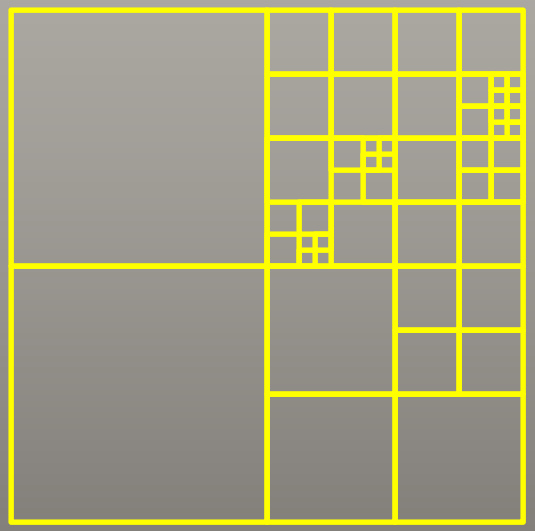


(example with K=3, dim=2)

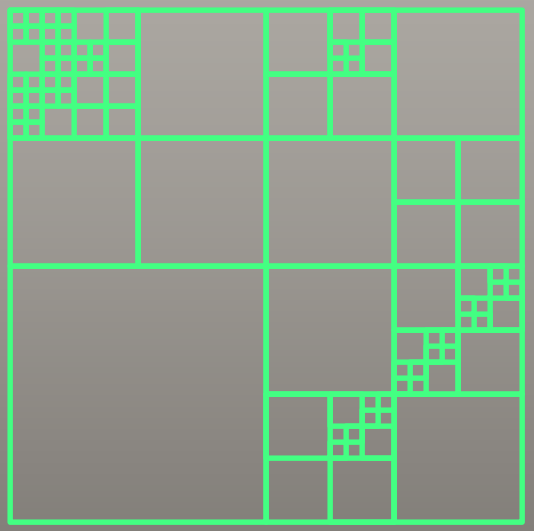
0.834	0.592	0.003
0.592	0.003	0.010
0.003	0.010	0.007

- We have to sample these polynomials to create a mesh
  - Can sample as finely as needed inside a file format reader
  - Each tree node has K^{dim} coefficients
  - K can be up to 20 (i.e. 400 coeffs/tree node in 2D, 8000 in 3D)
  - K^{dim} samples (1 per coeff) results in no loss of information

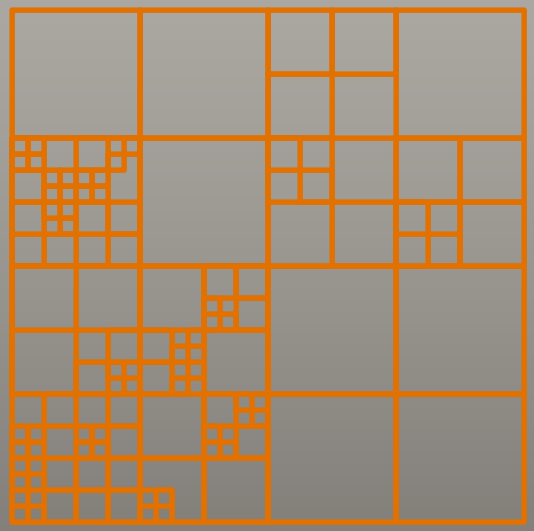
Each variable is refined independently



"*pressure*"

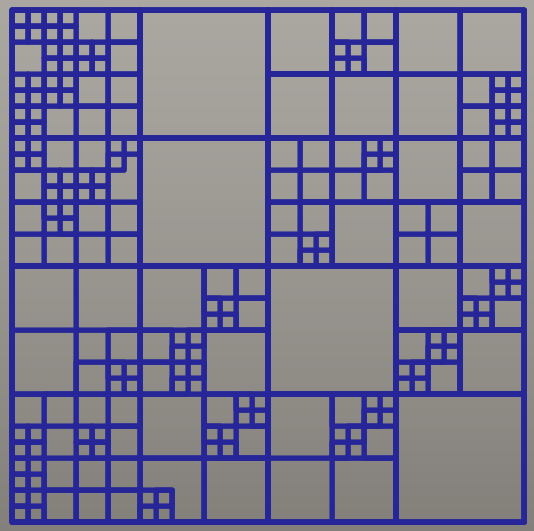


"*density*"



"*temperature*"

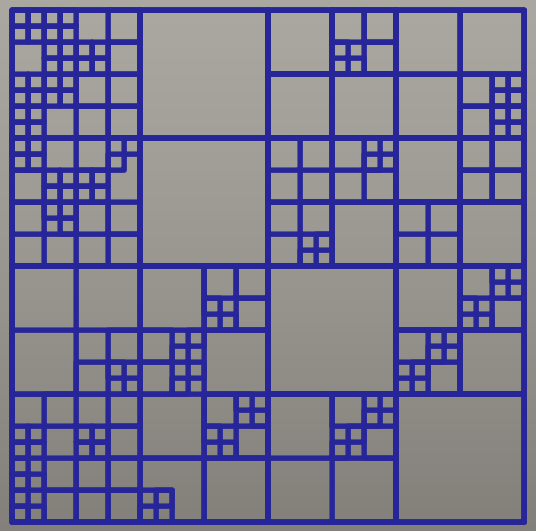
How to combine variables?



pressure × *density* ÷ *temperature*

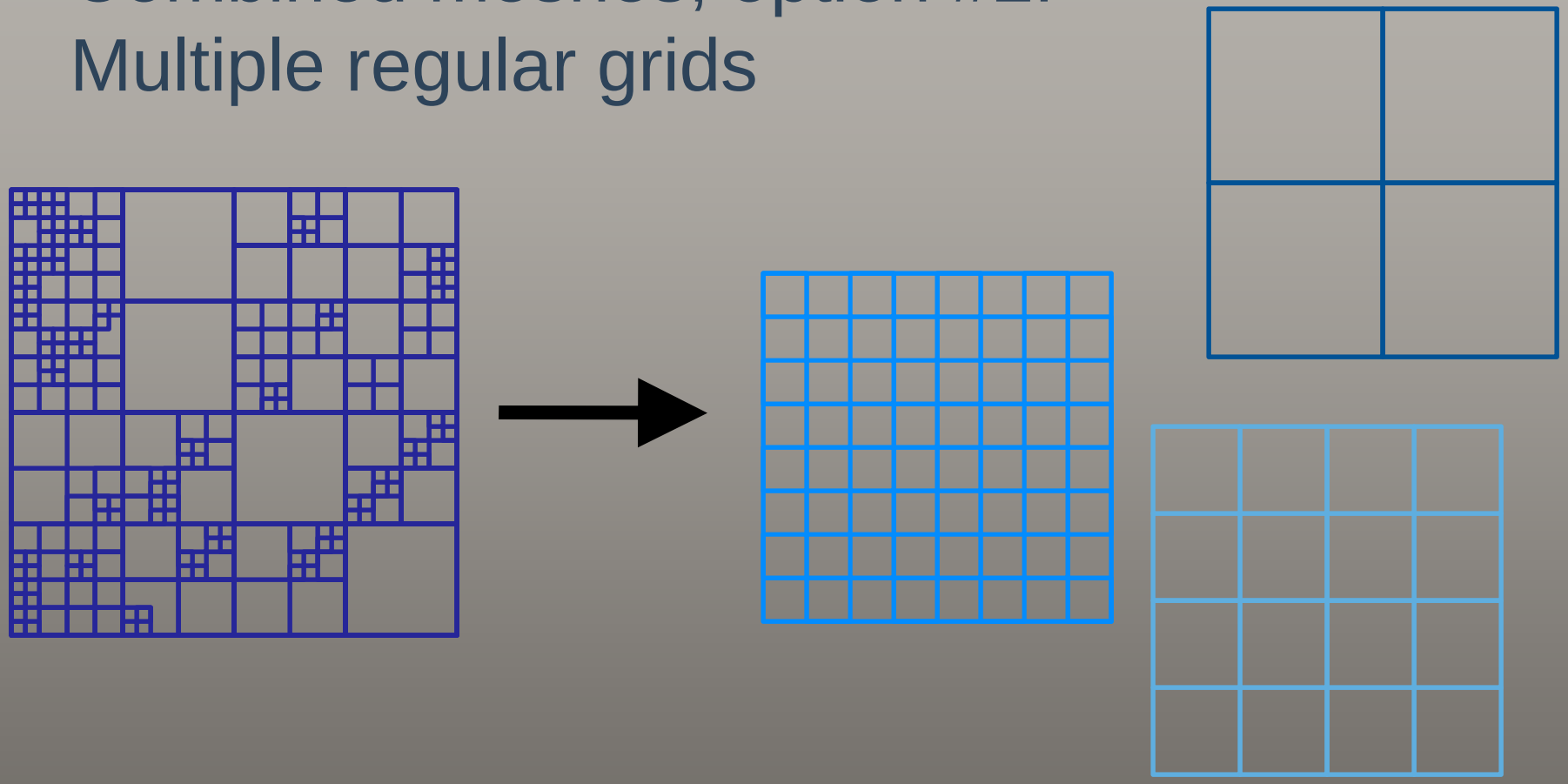
- Necessary for many expressions, operators, and other analysis

Combined meshes, option #1: Direct quadtree mesh union



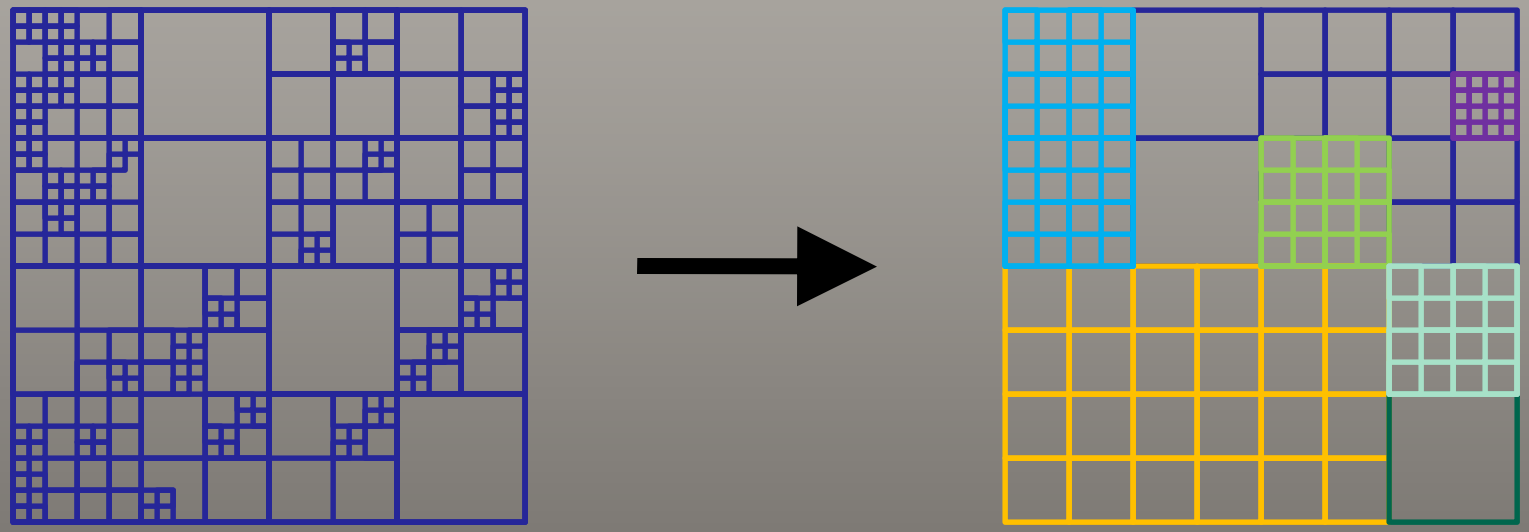
- Straight combination of *N* meshes
  - results in *N* × {#cells in one mesh}

Combined meshes, option #2: Multiple regular grids



- Pool of regular grids at selectable resolutions
- Up to $(2^{30})^3$ for full refinement (unfeasibly large)

Combined meshes, option #3: Unified block-structured AMR



- Heuristic combination to define regular patch levels

Further issues

- How to combine variables?
  - Hybrid approaches may be the only option
- Which coefficients are most efficient?
  - Legendre (more costly evaluation, but native)
  - Interpolating (simpler evaluation)
  - Wavelet (can evaluate at higher-level nodes)
- Parallelism
  - Free to choose our own decomposition
- File format
  - In on the ground floor, can advise/recommend scalable, parallel I/O layers like HDF5

Current Status

- Loading raw MADNESS tree
- Reading native Legendre coefficients
- Two meshes per variable
  - Unstructured tree for showing native internal data
  - Regular sampled grid for combined operations
- Working with code team on file formats, layout